Enhancing Distribution System Resilience: A First-Order Meta-RL algorithm for Critical Load Restoration

Zain ul Abdeen[1], Xiangyu Zhang[2], Waris Gill[1], Ming Jin[1]

***Abstract*— The increasing frequency of extreme events and the integration of distributed energy resources (DERs) into modern grids have elevated the need for resilient and efficient critical load restoration strategies in distribution systems. However, the strong inter-temporal dependency, stochastic nature of renewable DERs, and limited resources availability make the problem challenging. Although reinforcement learning (RL) and warm-start RL methods have shown promising results, their performance often falls short in rapidly adapting to new, unseen situations and typically requires exhaustive problem-specific tuning. To address these gaps, we propose a First-Order Meta-based RL (FOM-RL) algorithm within-online framework for adaptive and robust critical load restoration. Focusing on local DERs as the enabling technology, FOM-RL allows the RL agent to swiftly adapt to new unseen scenarios by leveraging previously acquired knowledge of different tasks. Experimental results provide evidence that proposed algorithm learns more efficiently and showcases generalization capabilities across diverse set of operational scenarios. Furthermore, we present a theoretical analysis yielding a tight sublinear regret bound that responds to temporal variability. The results, characterized by a task-averaged optimality gap bounded by $\mathcal{O}\left(\frac{V_M+D^*}{\sqrt{T}M}\right)$, suggest that optimality improves with task similarity and a greater number of tasks M.**

I. Introduction

The availability of reliable and quality electricity is an essential requirement for the smooth functioning of our technologically driven society. However, unforeseen events such as natural disasters, equipment failures, or cyber-attacks can lead to large-scale blackouts [1]–[3]. In such scenarios, the restoration of power to the critical loads – sectors where power interruption may cause catastrophic consequences for the security, society, and economy – is of utmost importance [4], [5]. However, the critical load restoration (CLR) problem in power systems is fraught with complexity, involving considerations such as load prioritization, generation capacity, and network constraints.. These complexities necessitate the exploration of novel methodologies that are adaptive, efficient, and effective in providing optimal solutions for the CLR problem.

Traditional methods for solving the CLR problem, such as optimization-based algorithms [6], heuristic methods [8], and deterministic techniques [9], have been widely used and have their merits. However, they may face challenges in handling the dynamic and non-linear nature of the power system during emergencies [10]. Strategies like DERs and microgrids [14], as well as methods that treat CLR as chance-constrained stochastic programming [7] or model predictive control (MPC) [9], have shown promise, but may need further adaptations to effectively handle the evolving context of climate-change-induced extreme events [11], [13].

Reinforcement learning has emerged as a complementary approach to optimization-based methods, offering the ability to learn optimal policies from interactions with the dynamically changing environment [15]. RL adaptability to non-linear and dynamic systems, without being restricted to specific model requirements, makes it a potentially powerful tool for the CLR problem [12]. However, the efficiency of model-free RL approaches in evolving contexts is an area that requires further investigation [13]. RL approaches face challenges due to the non-convex nature of the policy optimization landscape and the data-intensive training process. Techniques like curriculum-based RL [11] have been proposed to streamline the training process and enhance efficiency of RL agent training for CLR problems. While these methods have shown improvements, there is still room for further advancements in policy fine-tuning and generalization to new scenarios.

Meta-based RL algorithms, aim to create a generalized policies that can quickly adapt to new tasks [19], [20]. However, traditional Meta-RL frameworks like Model-Agnostic Meta-Learning are computationally demanding because they require second-order derivatives to update the Meta-parameters. These updates, necessitating Hessian matrix computations, can hinder efficiency by amplifying computational costs, especially when faced with the need for numerous gradient steps during the test phase. To address these limitations, we introduce a first-order Meta-RL framework [26], combined with Evolution-Strategy RL (ES-RL) [23], which simplifies the learning process by avoiding complex Hessian computations and employing gradient averaging. Gradient averaging involves updating the model parameters by averaging the first-order gradients obtained from task-specific optimizations, which significantly reduces the variability and instability in updates. This method not only curtails computational demands but also boosts generalization and adaptability, making it suitable for CLR problem.

Contribution: The key contributions of this study can be summarized as follows:

- A FOM-RL within-online framework is introduced to expedite the training of the CLR controller, thereby aiding it in converging to a more robust control policy and adaptable to a variety of tasks.

[1] The Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA, USA. Emails: {zabdeen,waris, jinming}@vt.edu

[2] National Renewable Energy Laboratory, Golden, CO, USA. Email:{Xiangyu.Zhang,}@nrel.gov

- We show that the task averaged regret for optimality gap decay sublinearly with respect to both the number of tasks M and steps T. Specifically we establish the bound of order $\mathcal{O}\left(\frac{V_M+D^*}{\sqrt{T}M}\right)$.

The structure of paper is organized as follows: In Section II, we present the formulation of the CLR problem. Section III elaborates on our proposed FOM-RL strategy designed for tackling the CLR problem. Theoretical analysis and regret bounds are the focus of Section IV, while Section V offers case study results to substantiate our approach. Lastly, Section VI concludes the paper with potential directions for future research.

II. Problem Formulation

A. Critical Loads Restoration Problem

In this paper, we examine multi-step prioritized CLR problem when a distribution system is isolated from the main grid. The objective is to optimize the restoration of prioritized critical loads to enhance system resilience throughout the outage, over discrete time intervals denoted by $\mathcal{T} = \{1, 2, \ldots, T\}$, using local DERs. Available DERs categorized into: dispatchable fuel-based ($\mathcal{D}^f$), dispatchable battery storage energy ($\mathcal{D}^s$), and renewable-based energy ($\mathcal{R}$). Each critical loads $i \in \mathcal{L}$ is assigned an importance factor ς^i, forming the priority vector $\boldsymbol{\varsigma} = [\varsigma^1, \varsigma^2, \ldots, \varsigma^N]^\top \in \mathbb{R}^N$. At each control step $t \in \mathcal{T}$, the controller establishes the active power set $\mathbf{p}_t^{\mathcal{G}} \in \mathbb{R}^{|\mathcal{G}|}$, and power factor angles $\alpha_t^{\mathcal{G}} \in \mathbb{R}^{|\mathcal{G}|}$ for all DERs (i.e. $\mathcal{G}$=$\mathcal{R} \cup \mathcal{D}^f \cup \mathcal{D}^s$). Concurrently, the controller defines the restored power levels for each critical load in terms of both active and reactive power, represented as $\mathbf{p}_t = [p_t^1, p_t^2, \ldots, p_t^N]^\top \in \mathbb{R}^N$ and $\mathbf{q}_t = [q_t^1, q_t^2, \ldots, q_t^N]^\top \in \mathbb{R}^N$. The objective is to optimize control function, as outlined in [11]:

$$\sum_{t\in\mathcal{T}} \left(\boldsymbol{\varsigma}^\top \mathbf{p}_t - \mu\boldsymbol{\varsigma}^\top [\mathbf{p}_{t-1} - \mathbf{p}_t]^+ + \nu_t\right) \tag{1}$$

where, $\nu_t := -\lambda\|[\boldsymbol{\nu}_t - \bar{\boldsymbol{\nu}}]^+ + [\underline{\boldsymbol{\nu}} - \boldsymbol{\nu}_t]^+\|_2^2$ represents the penalty for single step voltage violation across all N_b buses. Here, $\boldsymbol{\nu}_t \in \mathbb{R}^{N_b}$ denotes the vector of bus voltage magnitudes at time t, $\underline{\boldsymbol{\nu}} = V^{\min}1_{N_b} \in \mathbb{R}^{N_b}$, and $\bar{\boldsymbol{\nu}} = V^{\max}1_{N_b} \in \mathbb{R}^{N_b}$, represent the lower and upper voltage limits, respectively. The coefficient λ, encourages maintaining bus voltages to be within limits. It is worth noting that voltage bounds are included as a penalty term, as they represent system-controlled outcomes that can not be directly constrained within the framework of RL. The second term in (1) penalizes frequent load restoration and shedding by factor μ to provide a reliable and monotonic load restoration, thus mitigating the impact of fluctuating renewable energy sources. The first term, $\boldsymbol{\varsigma}^\top \mathbf{p}_t$ aims to optimize load restoration over time based on priority rankings, thereby enhancing the resilience of the system.

To optimize the objective function (1), certain operational constraints must be met for each time step $t \in \mathcal{T}$. Constraint (2) specifies the permissible power output ranges for micro-turbines, dictated by fuel availability; where, τ is the control interval and E^g is the known fuel reserve limit. Battery energy constraints (3)-(5) cover state of charge (SOC) bounds and charge/discharge rates, incorporating storage efficiency η_t, taking values based on whether the battery is charging $\eta_t = \eta^{\text{ch}}\left(\text{i.e.}, p_t^\theta > 0\right)$ or discharging $\eta_t = \frac{1}{\eta^{\text{dis}}}\left(\text{i.e.}, p_t^\theta < 0\right)$. S_t^θ and s_0 are the current and initial SOC. Renewable-based DER constraints are articulated in (6), where, $\hat{p}_t^r$ is the fluctuating renewable energy influenced by natural conditions and prioritized for use during restoration. Notably, the symbol $\hat{\cdot}$ signifies forecasted values. Finally, (7) delineates the feasible options for load restoration decisions.

$$p_t^g \in [\underline{p}^g, \bar{p}^g], \quad \alpha_t^g \in [\underline{\alpha}^g, \bar{\alpha}^g], \quad \sum_{t\in\mathcal{T}} p_t^g.\tau \leq E^g \tag{2}$$

$$-p^{\theta,\text{ch}} \leq p_t^\theta \leq p^{\theta,\text{dis}} \tag{3}$$

$$S_{t+1}^\theta = S_t^\theta - \eta_t \cdot p_t^\theta \cdot \tau, \quad S_0^\theta = s_0 \tag{4}$$

$$\underline{S}^\theta \leq S_t^\theta \leq \bar{S}^\theta, \quad \forall\ \theta \in \mathcal{D}^s \tag{5}$$

$$p_t^r = \hat{p}_t^r, \quad \alpha_t^r \in \left[\underline{\alpha}^r, \bar{\alpha}^r\right], \quad \forall r \in \mathcal{R} \tag{6}$$

$$\mathbf{0} \leq \mathbf{p_t} \leq \mathbf{p} \tag{7}$$

Integrating both the objective function and associated constraints, the optimal control problem for distribution system restoration is defined as:

$$\begin{aligned} &\text{maximize}_{\mathbf{p}_t, \mathbf{q}_t, \mathbf{p}_t^{\mathcal{G}}, \boldsymbol{\alpha}_t^{\mathcal{G}}, \forall t\in\mathcal{T}} \quad (1) \\ &\text{subject to}_{\forall t\in\mathcal{T}} \quad (2)-(7) \end{aligned} \tag{8}$$

The problem is framed as a Mixed-Integer Linear Programming (MILP). The existing methodologies for solving the MILP formulation of CLR problem includes like No Reserve MPC (NR-MPC) [9] and Reserve Considered MPC (RC-MPC) [28]. NR-MPC iteratively solves the MILP in a receding horizon manner, updating with new renewable forecasts, while RC-MPC incorporates a penalty for generation reserve to hedge against renewable forecast errors, enhancing robustness. However, these MILP solutions face limitations such as computational complexity, poor scalability in dynamic environments, and the inherent rigidity of linear programming which struggles with variable conditions. To address these challenges, in the subsequent sections III, we introduce a Meta-RL strategy. This approach leverages learning-based methods to adaptively handle the complexities associated with the CLR problem, including handling variability in DER outputs and operational uncertainties.

B. Sequence of CLR Problems

In the context of Meta-learning, we consider a sequence of CLR problems, indexed by $m = 1, \ldots, M$. Each problem in the sequence represents a distinct environment or scenario, driven by varying parameters such as critical load demands $(\mathbf{p}^{(m)}, \mathbf{q}^{(m)})$. These variations create a diverse set of CLR problems, each reflecting unique operational challenges. The goal of Meta-learning is to learn a policy that can quickly adapt to new CLR problems drawn from sequence, by leveraging the experience gained from solving previous problems. This formulation allows us to study the adaptability and generalization capabilities of the proposed methods in the face of varying environmental conditions and problem parameters.

Simulation and Evaluation: To evaluate the proposed methods for the CLR problem, we conduct simulations based on the following setup: 1) The demand for critical loads, represented as $\mathcal{L}$ ($\mathbf{p} = [p^1, \ldots, p^N]^\top \in \mathbb{R}^N$ and $\mathbf{q} = [q^1, \ldots, q^N]^\top \in \mathbb{R}^N$), is assumed to stay constant over the course of the outage, with partial restoration allowed at each time step. 2) Generation from photovoltaic (PV) and wind turbines can be forecasted, although these forecasts are not entirely precise, thereby providing a realistic model. 3)The simulations focus on the steady-state dispatch of DERs and load restoration decisions, neglecting the transient dynamics of the distribution system. 4) At the initial time step, the distribution network is assumed to be re-energized, reconfigured, and DERs are synchronized, with a stable topology throughout the restoration period. This assumption allows us to focus on the scheduling of DERs after the reconfiguration is complete but may not capture the full complexity of the restoration process.

III. Meta RL for Sequential CLR

In this section, we present how to use the FOM-RL algorithm to solve the above mentioned optimal control problem (8). This necessitate to reformulate the problem (8) into RL Markov decision process (MDP), and then show how to use FOM-RL to efficiently train a CLR controller.

A. CLR in RL Framework

Reformulating the optimal control problem (8) as a MDP allows us to leverage the powerful framework of RL to learn optimal control policies through interactions with the environment. This reformulation enables us to effectively handle the complexities, uncertainties, and sequential nature of the CLR problem, and develop adaptive control strategies that can generalize to new scenarios. State, action and reward are the key elements of MDP corresponding to the optimal control problem defined below.

State $(\mathcal{S})$: The state $s_t \in \mathcal{S}$ serve as the input to the policy for decision-making at each control step. It reflects system status at the current step and defined as:

$$s_t := \left[(\hat{p}_t^r)^\top, (\tilde{p}_{t-1})^\top, (S_t^\theta)^\top, (E_t^\mu)^\top, t\right]^\top \in \mathcal{S},$$

where, $\hat{p}_t^r$ is the renewable generation for the next hour, $\tilde{p}_{t-1} := \operatorname{diag}\{\mathbf{p}\}^{-1}\mathbf{p}_{t-1} \in \mathbb{R}^N$ shows the fractional load restoration level. S_t^θ is the state of charge, and E_t^μ denotes the fuel available for the micro-turbine, which indicates its residual capacity to support load. t represents the current time step index to inform progress.

Action $(\mathcal{A})$: At each control step $t \in \mathcal{T}$, the action a_t encapsulate the strategic decisions based on the current state determined by the RL policy i.e., $a_t = \pi(s_t; \phi)$. Specifically, a_t, is responsible for determining both the quantity of load to be restored and the active/reactive power outputs from selected DERs at each time step t, and formally defined as:

$$a_t := \left[(\mathbf{p}_t)^\top, \left(\mathbf{H}_p \mathbf{p}_t^{\mathcal{G}}\right)^\top, \left(\mathbf{H}_\alpha \boldsymbol{\alpha}_t^{\mathcal{G}}\right)^\top\right] \in \mathcal{A},$$

where, $\mathbf{H}_p \in \mathbb{R}^{(|\mathcal{D}^f|+\mathcal{D}^s|-1)\times|\mathcal{G}|}$ and $\mathbf{H}_\alpha \in \mathbb{R}^{(|\mathcal{G}-1|)\times|\mathcal{G}|}$ are selection matrices highlight the control over both the active power output from dispatchable DERs and the reactive power from all DERs. Here, $\mathbf{p}_t$ denotes the decision regarding the amount of load to restore, illustrating a targeted approach in managing the dynamic nature of grid restoration and DERs management.

Reward: r_t serves as a scalar evaluation of the control action a_t, based on the state s_t, i.e., $r_t = R(s_t, a_t)$. Corresponding to (1), the reward is defined as $r_t = \boldsymbol{\varsigma}^\top \mathbf{p}_t - \mu\boldsymbol{\varsigma}^\top[\mathbf{p}_{t-1} - \mathbf{p}_t]^+ + \nu_t$, and is computed using the simulation outcomes at t.

Within-task RL Training: In essence, the aim of RL agent is to find an optimal control policy $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})$ parametrized by ϕ, which maximizes the expected total reward $F(\phi) = \mathbb{E}_{\pi(s_t;\phi)}\left(\sum_{t\in\mathcal{T}} r_t\right)$. $F(\phi) = \mathbb{E}_{\pi(s_t;\phi)}\left(\sum_{t\in\mathcal{T}} r_t\right)$. We use $\Delta(\mathcal{A})^{|\mathcal{S}|}$ to denote the simplex over all states In our model, a neural network is used as the policy network. Policy based RL algorithms updates their parameters via gradient ascent, necessitating the computation of the policy gradient $\nabla_\phi F(\phi)$. A significant challenge in RL is the absence or inaccessibility of derivatives for the environment or policy functions. To circumvent the limitations of gradient-based estimation, we implements ES-RL, a gradient free optimization algorithm. ES-RL uses a population distribution $\mathcal{X}_\varphi$, modeled as an isotropic Gaussian with mean φ and fixed covariance $\sigma^2 I$. This allow to write $\mathbb{E}_{\phi\sim\mathcal{X}_\varphi} F(\phi)$=$\mathbb{E}_{\epsilon\sim N(0,I)} F(\phi + \sigma\epsilon)$, facilitating optimization directly over parameter ϕ using gradient ascent with the score function estimator: $\nabla_\phi \mathbb{E}_{\epsilon\sim N(0,I)} F(\phi+\sigma\epsilon) = \frac{1}{\sigma}\mathbb{E}_{\epsilon\sim N(0,I)} F(\phi+\sigma\epsilon)\epsilon$.

Unlike gradient-based RL methods such as PPO [24], TRPO [22], and DDPG [25], that follow the gradient of the expected reward, which can get ensnared by local maxima, ES-RL leverages a global search heuristic due to its stochastic nature. It has the ability to maintain and explore a diverse set of solutions, which increases the probability of escaping local optima.

B. Proposed FOM-RL Framework

The FOM-RL algorithm consists of two main stages: Meta-training and Meta-testing. During Meta-training, the algorithm aims to learn an initialization for the policy parameters that can quickly adapt to new tasks with minimal fine-tuning. This is achieved by exposing the algorithm to a diverse set of training tasks, each representing a different CLR problem.

For each task m, the policy parameters are initialized with the current Meta-parameters $\phi_{m,0}$ and then fine-tuned using ES-RL for a fixed number of iterations T. This within-task training allows the policy to adapt to the specific characteristics of each task. The Meta-parameters are then updated based on the cumulative performance across all tasks, using a first-order approximation of the Meta-gradient. This Meta-update step allows the algorithm to accumulate knowledge across tasks and learn a generalized initialization $\hat{\phi}_{m,j}$, that can rapidly adapt to new tasks.

In the Meta-testing stage, the learned Meta-parameters $\hat{\phi}_{m,j}$ serve as an initialization for the policy when faced with a new, unseen CLR problem. The policy is then fine-tuned using ES-RL for a small number of iterations, leveraging the knowledge gained during Meta-training to quickly adapt to the specific challenges of the new task.

The key steps of the FOM-RL algorithm, as outlined in Algorithm 1, provide a high-level overview of the process. The two-level structure of the FOM-RL algorithm, with ES-RL for within-task training and FOM-RL for across-task adaptation, offers several benefits. It allows the algorithm to leverage the strengths of ES-RL, such as its resilience to local optima and ability to maintain a diverse set of solutions, while also enabling fast adaptation to new tasks through the Meta-learning process. This combination of robustness, adaptability, and generalization makes the proposed approach well-suited for tackling the challenges of the CLR problem in various scenarios.

Algorithm 1: FOM-RL

```
Input: A set of $M$ tasks for Meta-training, number of
       iterations $T$, learning rate $\alpha$ for task-specific
       training, Meta-stepsize $\epsilon$.
Output: Meta-policy $\pi_M$ and optimal policy $\hat{\pi}_M$.
Function Train():
    Initialize random policy $\pi_{1,0}$ with parameters $\phi_{1,0}$
    for each task ($m = 1, \dots, M$) do
        Load initial policy with parameter $\phi_{m,0}$ for task $m$

        for iteration ($j = 1, \dots, T$) do
            Update policy using ES-RL
            Save the best model $\hat{\pi}_{m,j}$ and model
              parameter $\hat{\phi}_{m,j}$
        Meta-update:
          $\phi_{m+1,0} \leftarrow \phi_{m,0} + \frac{\epsilon}{M}\sum_{m=1}^{M}(\hat{\phi}_{m,j} - \phi_{m,0})$
    Save Meta-policy $\pi_M = \hat{\pi}_{M,j}$
Function Testing():
    Load Meta-policy $\pi_M$
    Fine-tune Meta-policy at test time on the new unseen
      task to receive the optimal policy $\hat{\pi}_M$
```

IV. Regret Bound for Meta-RL within-Online Framework

In this section, we present the theoretical analysis of the proposed algorithm. We examine MDPs where RL tasks emerge sequentially, indexed by $m = 1, \dots, M$. In each task m, the agent is required to iteratively refine the policy parameter $\{\phi_{m,j}\}_{j=0}^{T}$over T steps using ES-RL algorithm to receive suboptimal policy parameters $\hat{\phi}_{m,T}$. The following theorem provide convergence guarantees to ES-RL.

Theorem 4.1 (Theorem 6; [18]): Suppose ES-RL policy update for each task m, perform $T = \frac{4(N+4)^2L^2R^2}{\epsilon^2}$ iterations with learning rate α_m=$\frac{R}{(N+4)(T+1)^{1/2}L}$ to optimize objective function $F_m(\cdot)$. For $\sigma \leq \frac{\epsilon}{2LN^{1/2}}$, then the sub-optimality gap for each task m is bounded by;

$$\mathbb{E}\left[F_m\left(\hat{\phi}_{m,T}\right)\right] - F_m\left(\phi_m^*\right) \leq \frac{2(N+4)L\|\phi_m^* - \phi_{m,0}\|}{T^{1/2}}, \tag{9}$$

where ϕ_m^* are the parameters of optimal policy π_m^*, R being the bound of $\|\phi_m^* - \phi_{m,0}\| \leq R$.
We can observe from (9), that the regret bound is depending on the parameters of the policy initialization. Beyond the single task, Meta-algorithm aims to sequentially update the initial policy $\pi_{m,0}$ parameters $\phi_{m,0}$. Therefore we aim to minimize the task average optimality gap (TAOG). In Meta-RL, the extent to which TAOG improves is influenced by the similarity among the sequential MDP tasks [20]. For any fixed initial policies parameters $\{\phi\}$, the task similarity can be measured by $D^* = \min_{\phi\in\Delta(\mathcal{A})^{|\mathcal{S}|}} \frac{1}{M}\sum\limits_{m=1}^{M}\|\phi_m^* - \phi\|$. The following theorem shows the TAOG for the proposed MDP-within-online framework.

Theorem 4.2 (Task Average Optimality Gap): Let $\{\phi_{m,0}\}_{m=0}^{M}$ be the initialization for each task determined by follow the average leader. For each task we train the policy for T steps with learning rate α and obtain $\left\{\hat{\phi}_{m,T}\right\}_{m=1}^{M}$. Let ϕ_m^* is the optimal Meta initialization for each task, then the task average optimality gap is bounded as

$$\frac{1}{M}\sum_{m=1}^{M}\mathbb{E}\left[F_m(\hat{\phi}_{m,T})\right] - F_m(\phi_m^*) \leq \mathcal{O}\left(\frac{V_M + D^*}{\sqrt{T}M}\right). \tag{10}$$

Proof: See link.... ■

Remark 1: The task-averaged regret upper bound is sensitive to temporal variability V_M. Specifically, a lower V_M results in a tighter bound, indicating the algorithm performs better in environments with stable, less variable tasks. A larger D^* loosens the upper bound, implying that as tasks become more dissimilar, the algorithm may become less effective at generalizing across these tasks.

Remark 2: The terms T and M in the denominator suggest that increasing the number of iterations per task T or the total number of tasks M could lead to a reduced regret. However, the square root indicates a sub-linear rate.

V. Case Study

A. Experiment Setup

The experiment is conducted on a modified IEEE-13 bus test system with 15 critical loads distributed in a three-phase system Fig. 1. Four DERs are considered, with parameters summarized in Table I. The control horizon for load restoration is set to six hours, with a control interval of five minutes, resulting in a total of 72 time steps. An OpenAI Gym [16] learning environment is used to interface the RL agent with the OpenDSS [17] grid simulator, handling RL-agent-grid interactions and enforcing operational constraints, including box constraints on individual power elements and power balance constraints to maintain system stability. Voltage limits are set to $[0.95, 1.05]$ p.u., with a penalty of $\lambda = 10^8$ for violations.

TABLE I: Parameters of DERs

DERs	Parameters
Energy Storage (ST)	$-P^{\theta,ch} = P^{\theta,dis} = 1200$ $160 \leq S_t^{\theta} \leq 1250, \quad \alpha^{\theta} \in [0.\pi/4]$
Micro-Turbine (MT)	$p^{\mu} \in [0, 400], \quad E^{\mu} = 1200$ $\alpha^{\mu} \in [0, \pi/4]$
Photovoltaic (PV)	$p^{\rho} \in [0, 300]\,, \quad \alpha^{\rho} \in [0, \pi/4]$
Wind Energy (WT)	$p^{\omega} \in [0, 300]\,, \quad \alpha^{\omega} \in [0, \pi/4]$

To train and evaluate the adaptability of the proposed Meta-RL method, a series of tasks with varying base load demand values ($\mathbf{p_t}$) are generated, creating diverse grid control scenarios.

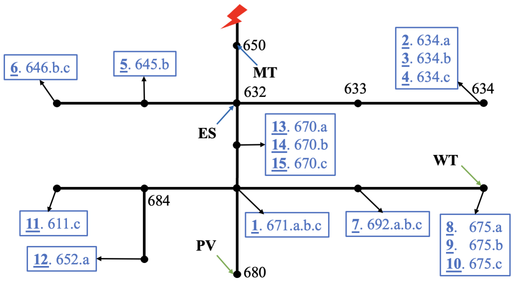

Fig. 1: Modified IEEE-13 bus system

B. Performance Comparison

In this section, we evaluate the efficacy of the proposed Meta-RL algorithm 1 by comparing it with ES-RL and warm-start RL algorithm 2. Warm-start RL is implemented by initializing each task's policy with the best policy from the previous task. Figure 2 illustrates the mean rewards against the number of training steps, showing that Meta-RL achieves higher mean rewards and maintains this advantage throughout the training. Warm-start RL benefits from pre-existing knowledge but achieves moderate gains, while ES-RL improves at a slower pace.

To quantify the performance of Meta-RL and warm-start RL, we use two metrics: Jump-start (Δ_{init}), indicating the immediate performance advantage, and asymptotic performance (ΔR :=difference of reward value from RL at the end of training), reflecting the reward improvement by the end of the learning steps. Both warm-start and Meta-policy (π_M) undergo sequential training over 22 tasks before being tested on scenarios 23 (for Δ_{init}) to 27 (for ΔR). Table II reveals that Meta-RL consistently offers an immediate advantage (positive Δ_{init}) and demonstrates superior long-term learning (robust ΔR). Warm-Start RL shows variable initial performance but achieves gains by the end of learning, though not as consistently as Meta-RL. In Task 26, Meta-RL significantly outperforms Warm-Start RL, highlighting its ability to adapt rapidly to new tasks while maintaining a trajectory of improvement.

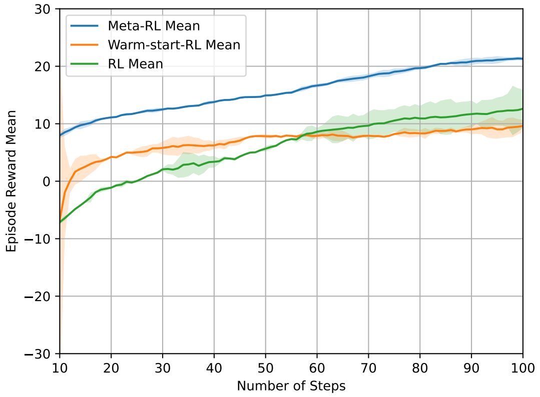

Fig. 2: Learning curves of Meta-RL, warm-start-RL and RL, illustrating Meta-RL superior efficacy in expected performance. The graph showcases the mean and variance of average rewards over 5 experimental runs.

```
Algorithm 2: Warm-Start-RL
Initialize: random policy $\pi_{1,0}$ with parameters $\phi_{1,0}$.
for each task $(m = 1, \ldots, M)$ do
    Load initial policy with parameter $\phi_{m,0}$ for task $m$.
    for iteration $(j = 1, \ldots, T)$ do
        Update policy using ES-RL.
    Save the best model $\hat{\pi}_{m,j}$ and model parameter $\hat{\phi}_{m,j}$.
    Set $\pi_{m+1,0} = \hat{\pi}_{m,j}$ and $\phi_{m+1,0} \leftarrow \hat{\phi}_{m,j}$.
Save warm-start-policy $\pi_M = \hat{\pi}_{M,j}$.
```

C. Controller Evaluation

To see the performance of trained controller we deployed it to the unseen environment. Figure 3 illustrates the restoration process of an unseen testing scenario, showcasing the load restoration sequence and DERs generation profile over a 6-hour horizon. The left-most figure reveals that loads are restored either wholly or to a feasible extent, depending on the availability of renewable generation. The restoration follows a pattern where more critical loads are prioritized. The middle graph indicates the fluctuating nature of renewable resources (PV and WT), the stabilizing influence of storage, and the responsiveness of dispatchable resources like micro-turbines. The controller effectively manages the intermittency of renewable generation sources by leveraging storage and micro-turbines, ensuring a continuous and orderly restoration of loads according to their priority. The right-most graphs in

TABLE II: Performance metrics

Task Id.	Meta-RL		Warm-Start RL	
	Δ_{init}	ΔR	Δ_{init}	ΔR
23	**42.02**	**7.439**	-33.04	3.839
24	19.89	**3.12**	**28.72**	0.17
25	**41.18**	**4.52**	-143.4	2.32
26	**89.09**	**46.13**	-99.23	39.07
27	13.23	9.38	**28.22**	**10.25**

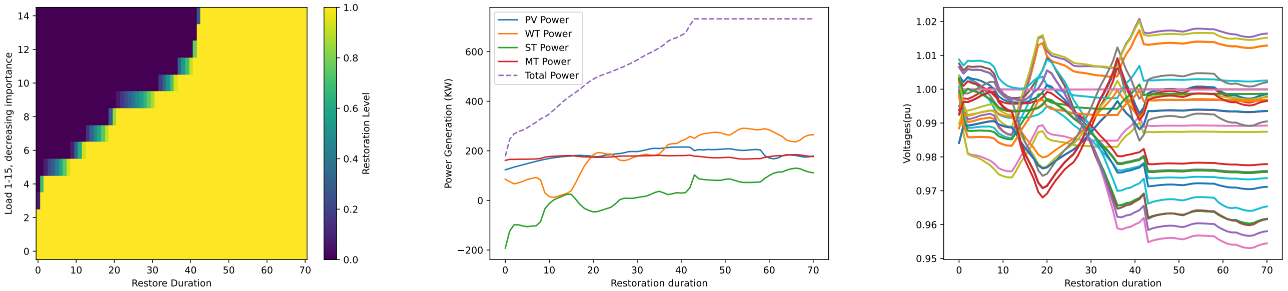

Fig. 3: Three sub-figures shows the load restoration profile total power and DERs generation profile, and all buses' voltage profiles during restoration, respectively.

Figure 3 ensure safe operation by keeping the voltages within the desired range of $[0.95, 1.05]$.

VI. Conclusion and Future Directions

In this work, we have demonstrated the Superior performance of the FOM-RL compared to RL and warm-start RL to learn better policy for CLR problem. The trained controller demonstrates superior adaptability to unforeseen events and reduces the necessity for extensive tuning. Our empirical findings confirm that FOM-RL generalizes across a spectrum of operational conditions, backed by a theoretical analysis that promises a tight sublinear regret bound. This translates to an improved optimality that scales with task similarity and the number of tasks involved, as reflected in the task-averaged optimality gap. The potential of FOM-RL to transform grid management practices through adaptive control and robust performance is profound, indicating a promising direction for future research in grid control applications.

References

[1] Gholami, Amin, et al. "Toward a consensus on the definition and taxonomy of power system resilience." IEEE Access 6 (2018): 32035-32053.

[2] U.S. Energy Information Administration, Reliability Metrics of U.S. Distribution System, Electric Power Annual 2021, Table 11.1.

[3] Panteli, Mathaios, and Pierluigi Mancarella. "Influence of extreme weather and climate change on the resilience of power systems: Impacts and possible mitigation strategies." Electric Power Systems Research 127 (2015): 259-270.

[4] Salman, Abdullahi M., Yue Li, and Mark G. Stewart. "Evaluating system reliability and targeted hardening strategies of power distribution systems subjected to hurricanes." Reliability Engineering and System Safety 144 (2015): 319-333.3.

[5] Panteli, Mathaios, and Pierluigi Mancarella. "The grid: Stronger, bigger, smarter?: Presenting a conceptual framework of power system resilience." IEEE Power and Energy Magazine 13.3 (2015): 58-66.

[6] Liu, Wenxia, et al. "A bi-level interval robust optimization model for service restoration in flexible distribution networks." IEEE Transactions on Power Systems 36.3 (2020): 1843-1855.

[7] Gao, Haixiang, et al. "Resilience-oriented critical load restoration using microgrids in distribution systems." IEEE Transactions on Smart Grid 7.6 (2016): 2837-2848.

[8] Wang, Ying, et al. "Coordinating multiple sources for service restoration to enhance resilience of distribution systems." IEEE Transactions on Smart Grid 10.5 (2019): 5781-5793.

[9] Liu, Weijia, and Fei Ding. "Collaborative distribution system restoration planning and real-time dispatch considering behind-the-meter DERs." IEEE Transactions on Power Systems 36.4 (2020): 3629-3644.

[10] Huang, Yuxiong, et al. "Resilient distribution networks by microgrid formation using deep reinforcement learning." IEEE Transactions on Smart Grid 13.6 (2022): 4918-4930.

[11] Zhang, Xiangyu, et al. "Curriculum-based reinforcement learning for distribution system critical load restoration." IEEE Transactions on Power Systems (2022).

[12] Bedoya, Juan Carlos, Yubo Wang, and Chen-Ching Liu. "Distribution system resilience under asynchronous information using deep reinforcement learning." IEEE Transactions on Power Systems 36.5 (2021): 4235-4245.

[13] Zhao, Tianqiao, and Jianhui Wang. "Learning sequential distribution system restoration via graph-reinforcement learning." IEEE Transactions on Power Systems 37.2 (2021): 1601-1611.

[14] Xu, Yin, et al. "Microgrids for service restoration to critical load in a resilient distribution system." IEEE Transactions on Smart Grid 9.1 (2016): 426-437.

[15] Du, Yan, and Di Wu. "Deep reinforcement learning from demonstrations to assist service restoration in islanded microgrids." IEEE Transactions on Sustainable Energy 13.2 (2022): 1062-1072.

[16] Brockman, Greg, et al. "Openai gym." arXiv preprint arXiv:1606.01540 (2016).

[17] Dugan, Roger C., and D. Montenegro. "The open distribution system simulator (OpenDSS): Reference guide." Electric Power Research Institute (EPRI) (2018).

[18] Nesterov, Yurii, and Vladimir Spokoiny. "Random gradient-free minimization of convex functions." Foundations of Computational Mathematics 17.2 (2017): 527-566.

[19] Finn, Chelsea, Pieter Abbeel, and Sergey Levine. "Model-agnostic Meta-learning for fast adaptation of deep networks." International conference on machine learning. PMLR, 2017.

[20] Khattar, Vanshaj, et al. "A CMDP-within-online framework for Meta-safe reinforcement learning." The Eleventh International Conference on Learning Representations. 2022.

[21] Zinkevich, Martin. "Online convex programming and generalized infinitesimal gradient ascent." Proceedings of the 20th international conference on machine learning (icml-03). 2003.

[22] Schulman, John, et al. "Trust region policy optimization." International conference on machine learning. PMLR, 2015.

[23] Salimans, Tim, et al. "Evolution strategies as a scalable alternative to reinforcement learning." arXiv preprint arXiv:1703.03864 (2017).

[24] Schulman, John, et al. "Proximal policy optimization algorithms." arXiv preprint arXiv:1707.06347 (2017).

[25] Lillicrap, Timothy P., et al. "Continuous control with deep reinforcement learning." arXiv preprint arXiv:1509.02971 (2015).

[26] Nichol, Alex, Joshua Achiam, and John Schulman. "On first-order Meta-learning algorithms." arXiv preprint arXiv:1803.02999 (2018).

[27] Campolongo, Nicolo, and Francesco Orabona. "Temporal variability in implicit online learning." Advances in neural information processing systems 33 (2020): 12377-12387.

[28] Eseye, Abinet Tesfaye, et al. "Resilient operation of power distribution systems using MPC-based critical service restoration." 2021 IEEE Green Technologies Conference (GreenTech). IEEE, 2021.

VII. Appendix

A. preliminaries

Before presenting the proofs of our main results, in this section, we establish necessary definitions and notational conventions. Let $f : \mathbb{R}^d \to \mathbb{R}$ be a function, the sub-gradient of f at a point x is denoted by $\partial f(x)$. We say that f is μ-strongly convex over a convex set $V \subseteq \text{int dom}(f)$ with respect to a norm $\|\cdot\|$ if, for any $x, y \in V$ and $g \in \partial f(x)$, it holds that $f(y) \geq f(x) + \langle g, y - x\rangle + \frac{\mu}{2}\|x - y\|^2$. Moreover, define $\psi : \mathcal{X} \to \mathbb{R}$ as a strictly convex and continuously differentiable function on int$\mathcal{X}$. The Bregman Divergence associated with ψ is given by $B_\psi(x, y) = \psi(x) - \psi(y) - \langle \nabla\psi(y), x - y\rangle$, assuming ψ is strongly convex with respect to the norm $\|\cdot\|$ on int$\mathcal{X}$.

B. Proof of Task-Average-Optimality-Gap for Meta-based RL Algorithm

This section provides a detailed proof of the Task-Average-Optimality-Gap for our proposed meta-based RL algorithm. We begin with an analysis of the ES-RL algorithm applied to a single task, initially establishing a regret bound.

1) Single Task Analysis: Consider a series of Markov Decision Processes, where RL tasks emerge sequentially, indexed by $m = 1, \ldots, M$. In each task m, the agent refines its policy parameter $\{\phi_{m,j}\}_{j=0}^T$ over T iterations using the ES-RL algorithm. We present the following theorem with convergence guarantees for the ES-RL algorithm:

Theorem 7.1 (Theorem 6; [18]): If the ES-RL policy updates for each task m perform $T = \frac{4(N+4)^2L^2R^2}{\epsilon^2}$ iterations with a learning rate $\alpha_m = \frac{R}{(N+4)(T+1)^{1/2}L}$, and if $\sigma \leq \frac{\epsilon}{2L\sqrt{N}}$, then the sub-optimality gap for each task m is bounded by:

$$\mathbb{E}\left[F_m\left(\hat{\phi}_{m,T}\right)\right] - F_m\left(\phi_m^*\right) \leq \frac{2(N+4)L\|\phi_m^* - \phi_{m,0}\|}{\sqrt{T}}, \tag{11}$$

where, ϕ_m^* represents the parameters of the optimal policy π_m^*, and R bounds $\|\phi_m^* - \phi_{m,0}\| \leq R$.

2) Extension to Multiple Tasks: Extending the single task analysis to multiple tasks within the meta-learning framework, the task average optimality gap across multiple tasks within the meta-learning framework is defined as:

$$\begin{aligned}\frac{1}{M}\sum_{m=1}^{M}\left[\mathbb{E}\left[F_m(\hat{\phi}_m)\right] - F_m(\phi_m^*)\right]\\ \leq \frac{2(N+4)L}{M\sqrt{T}}\sum_{m=1}^{M}\|\phi_m^* - \phi_{m,0}\|.\end{aligned} \tag{12}$$

The right side of inequality (12) shows that the task-averaged regret is upper bounded by terms based on parameter initialization $\phi_{m,0}$. As the meta-algorithm sequentially updates these initial parameters through online learning, it is expected to reduce the task average sub-optimality as more tasks are addressed. We can consider the right-hand side of (12) as an individual loss function (i.e., $l_m(\phi_{m,0}) := \|\phi_m^* - \phi_{m,0}\|$), allowing us to bound the dynamic regrets (i.e., TAOG), measured by a dynamic sequence of optimal policy parameters $\{\phi_m^*\}_{m=1}^M$, via static regret, which is measured against a fixed policy parameter ϕ.

3) Static Regret Analysis: In this section we provide the static regret bound, which are used to furnish the upper bound on TAOG of the proposed algorithm. The lemma below provides a bound on the static regret:

Lemma 7.1 ([27]): Assuming the domain of the loss function is a non-empty closed convex set and the Bregman divergence is γ-Lipschitz continuous with $D_b = \max_{a,b\in\text{Dom}(f)} B_\psi(a, b)$, let η_m be a non-increasing sequence. Employing implicit online mirror descent or Follow The Regularized Leader (FTRL) on a sequence of loss functions $\{l_m\}_{m=1}^M$ where $l_m(\phi_{m,0}) = \|\phi_m^* - \phi_{m,0}\|$, the static regret against a fixed comparator ϕ_0^* is bounded by:

$$\frac{1}{M}\sum_{m=1}^{M} l_m(\phi_{m,0}) - l_m(\phi_0^*) \leq \frac{D_b}{\eta_m M} + \frac{\sum_{m=1}^M \delta_m}{M}, \tag{13}$$

where $\delta_m = l_m(\phi_{m,0}) - l_m(\phi_{m+1,0}) - \frac{B_\psi(\phi_{m+1,0},\phi_{m,0})}{\eta_m}$.

Theorem 7.2 (Theorem 6.2; [27]): Under the assumptions of Lemma 7.1, and if η_m is a decreasing sequence, the average static regret is bounded by:

$$\begin{aligned}\frac{1}{M}\sum_{m=1}^{M} l_m(\phi_{m,0}) - l_m(\phi_0^*) \leq \frac{2}{M}\min\{\sqrt{\beta\sum_{m=1}^{M}\mathbb{E}_m\left[g_m^2\right]}\},\\ (l_1(\phi_{1,0}) - l_M(\phi_{M+1,0}) + V_M)\},\end{aligned} \tag{14}$$

where $V_M(f) = \sum_{m=2}^M \max_{\phi_{m,0}\in\text{Dom}(f)} |l_m(\phi_{m,0}) - l_m(\phi_{m-1,0})|$ is the temporal variability of the loss function. Note that the regret bound analyzed above is defined with respect to the optimal initial policy parameters ϕ_0^* in hindsight, not the final learned policy parameters.

C. Proof of Main Result

In Meta-RL, the extent to which TAOG improves is influenced by the similarity among the sequential MDP tasks [20]. For any fixed initial policies parameters $\{\phi\}$, the task similarity can be measured by $D^{*2} = \min_{\phi\in\Delta(\mathcal{A})^{|\mathcal{S}|}} \frac{1}{M}\sum_{m=1}^{M}\|\phi_m^* - \phi\|$. If the optimal policy parameter is not unique, we take the worst case for D^*, i.e., a set of policies for which D^{*2} is maximum. Building on the established foundations, we present a proof of the main theorem concerning the task average optimality gap:

Proof of Theorem ??

Proof: Define $\bar{R}$ as:

$$\bar{R} = \frac{1}{M}\sum_{m=1}^{M}\mathbb{E}\left[F_m(\hat{\phi}_m)\right] - F_m(\phi_m^*). \tag{15}$$

Given the bounds established in Theorem **??**, TAOG is further bounded by:

$$\begin{aligned}
\bar{R} &\le \sum_{m=1}^{M} \frac{2(N+4)L}{MT^{1/2}} (\|\phi_m^* - \phi_{m,0}\|) \\
&\overset{1}{=} \frac{2(N+4)L}{MT^{1/2}} \sum_{m=1}^{M} (\|\phi_m^* - \phi_{m,0}\|) - (\|\phi_m^* - \phi_0\|) + (\|\phi_m^* - \phi_0\|) \\
&\overset{2}{=} \frac{2(N+4)L}{MT^{1/2}} \sum_{m=1}^{M} (l_m(\phi_{m,0}) - l_m(\phi_0^*)) \\
&+ \frac{2(N+4)L}{MT^{1/2}} \sum_{m=1}^{M} (\|\phi_m^* - \phi_0\|) \\
&\overset{3}{\le} \frac{2(N+4)L}{MT^{1/2}} (l_1(\phi_{1,0}) - l_M(\phi_{M+1,0}) + V_M) \\
&+ \frac{2(N+4)L}{MT^{1/2}} \sum_{m=1}^{M} (\|\phi_m^* - \phi_0\|) \\
&\overset{4}{\le} \frac{2(N+4)L}{MT^{1/2}} (3D^* + V_M) .
\end{aligned} \tag{16}$$

In the above proof inequality 3 is directly follows from the results in (14), and last inequality 4 follows from the definition of task similarity index. This completes the proof, showing the bounded nature of TAOG under our meta-learning framework. ■